Claims

What is claimed is:

1. Apparatus for implementing enhanced graphical user interface functions in a graphical debugger comprising:

a user interface for operatively controlling a graphical user interface;

a loadmap display manager coupled to said user interface for implementing a loadmap function;

said user interface responsive to said loadmap display manager for displaying a program loadmap;

a custom record display manager coupled to said user interface receiving user inputs and implementing a custom record display function;

said user interface responsive to said custom record display manager for displaying user selected customized records.

2. Apparatus for implementing enhanced graphical user interface functions as recited in claim 1 further includes a debugger server and wherein said loadmap display manager operatively controls said debugger server for implementing said loadmap function.

3. Apparatus for implementing enhanced graphical user interface functions as recited in claim 2 wherein said load map function includes debugger means for reading debug data for a program under debug; examining the program debug data, for generating a list of each source and disassembly file contained in the program under debug, and for generating a list of each object or archive program bound to the program under debug at run time to generate said program loadmap for display.

4. Apparatus for implementing enhanced graphical user interface functions as recited in claim 3 wherein said load map function further includes debugger means for identifying program load and unload events, and for dynamically updating said program loadmap for display as loadmap information changes responsive to program load and unload events; whereby said program loadmap enables setting a breakpoint within a user selected address range for one instance of a source file, without setting the breakpoint in other instances of the source file.

5. Apparatus for implementing enhanced graphical user interface functions as recited in claim 2 wherein said custom record display manager operatively controls said debugger server for implementing said custom record display function; and said custom record display function includes debugger means for identifying a user selected variable, and user selected fields of the variable to be displayed.

6. Apparatus for implementing enhanced graphical user interface functions as recited in claim 5 wherein said custom record display function further includes debugger means for identifying a user selected all variables of this type to be customized, for creating a custom record with the user selected fields and for adding said created custom record to a custom type list for displaying user selected customized records for all variables of this type only with the user selected fields.

7. Apparatus for implementing enhanced graphical user interface functions as recited in claim 5 wherein said custom record display function further includes debugger means for creating a custom record with the user selected fields and for adding said created custom record to a variables list for displaying the variable only with the user selected fields.

8. A debugger computer program product for implementing enhanced graphical user interface functions in a computer system, said debugger computer program product including instructions executed by the computer system to cause the computer system to perform the steps of:

reading debug data for a program under debug;

examining said program debug data, generating a list of each source and disassembly file contained in the program under debug, and generating a list of each object or archive program bound to the program under debug at run time to generate said program loadmap for display; and

displaying said generated program loadmap on a graphical user interface.

9. A debugger computer program product for implementing enhanced graphical user interface functions as recited in claim 8 includes the step of identifying program load and unload events, and dynamically updating said program loadmap for display as loadmap information changes responsive to program load and unload events.

10. A debugger computer program product for implementing enhanced graphical user interface functions as recited in claim 9 wherein the step of identifying program load and unload events includes the steps of setting breakpoints at program load and unload entry points in the program under debug; and when debugging said program, identifying an unload breakpoint; and removing all information for an archive program being unloaded from said program loadmap to provide a current program loadmap, and displaying said current program loadmap.

11. A debugger computer program product for implementing enhanced graphical user interface functions as recited in claim 8 includes the step of identifying a user selected variable and identifying user selected fields of the variable to be displayed, and creating a custom record with the user selected fields.

12. A debugger computer program product for implementing enhanced graphical user interface functions as recited in claim 11 includes the step responsive to a user selection of all variables of this type to be customized, adding said created custom record to a custom type list for displaying user selected customized records for all variables of this type only with the user selected fields.

13. A debugger computer program product for implementing enhanced graphical user interface functions as recited in claim 11 includes the step adding said created custom record to a variables list for displaying the variable only with the user selected fields.

14. A debugger computer program product for implementing enhanced graphical user interface functions in a computer system, said debugger computer program product including instructions executed by the computer system to cause the computer system to perform the steps of:

identifying a user selected variable and identifying user selected fields of the variable to be displayed,

creating a custom record with the user selected fields; and

displaying the variable only with the user selected fields.

15. A debugger computer program product for implementing enhanced graphical user interface functions as recited in claim 14 wherein the step of displaying the variable only with the user selected fields includes the step adding said created custom record to a variables list for displaying the variable only with the user selected fields.

16. A debugger computer program product for implementing enhanced graphical user interface functions as recited in claim 14 includes the step of identifying a user selection of all variables of this type to be customized, adding said created custom record to a custom type list for displaying user selected customized records for all variables of this type only with the user selected fields.

17. A method for implementing enhanced graphical user interface functions in a graphical debugger comprising the steps of:

utilizing a loadmap display manager for implementing a loadmap function and generating a program loadmap for a program under debug;

displaying said generated program loadmap responsive to said loadmap function;

utilizing a custom record display manager for receiving user inputs and implementing a custom record display function;

displaying user selected customized records responsive to said custom record display function.

18. A method for implementing enhanced graphical user interface functions in a graphical debugger as recited in claim 17 wherein the step of utilizing a loadmap display manager for implementing a loadmap function and generating a program loadmap for a program under debug includes the steps of reading debug data for a program under debug; examining said program debug data, generating a list of each source and disassembly file contained in the program under debug, and generating a list of each object or archive program bound to the program under debug at run time to generate said program loadmap for display; and displaying said generated program loadmap on a debugger graphical user interface.

19. A method for implementing enhanced graphical user interface functions in a graphical debugger as recited in claim 18 includes the steps of identifying program load and unload events, dynamically updating said program loadmap and displaying said updated loadmap responsive to identified program load and unload events.

20. A method for implementing enhanced graphical user interface functions in a graphical debugger as recited in claim 17 wherein the step of utilizing a custom record display manager for receiving user inputs and implementing a custom record display function includes the steps of identifying a user selected variable and identifying user selected fields of the variable to be displayed, creating a custom record with the user selected fields; and displaying the variable only with the user selected fields.

21. A method for implementing enhanced graphical user interface functions in a graphical debugger as recited in claim 20 includes the steps of identifying a user selection of all variables of this type to be customized, adding said created custom record to a custom type list for displaying user selected customized records for all variables of this type only with the user selected fields.